

Chapter 15

Automation, Protections, and Tribulations

“On bare wings I brought them unto me.”

—from a plaque in an airmen’s memorial on the River Thames,
London; an adaptation from Exodus 19:4.

The long black runway sweated with haze as the elegant airliner made a swift landing. Within seconds, the wide-body aircraft reached the end of the runway and stopped. The regular stir of passengers—anxious to get out of their seats, eager to collect their belongings and disembark—was missing. In the large cabin, designed to carry more than 300 passengers, sat only three men. Two of them were seated side-by-side in the middle of the aircraft. In the forward section, behind a large console full of computer screens, panels with switches and buttons, rolls of printouts, and a mass of multi-colored wires, sat another man. He was busy flipping charts and moving switches while reading aloud into a microphone: “takeoff in flaps configuration 2, engage the autopilot at 157 knots and 6.5 seconds after takeoff, then reduce power on the right engine, shut hydraulic . . .”

This, apparently, was no ordinary flight.

Ahead of him was an empty galley, beyond it a lavatory, and then an aluminum door. Behind the door, a blue carpet led into a wide cockpit that looked like the flight deck on a starship. It was slick and clean, full of large displays and dozens of dazzling and colorful indicator lights. Above the displays was the long flight control panel, full of switches and knobs for engaging the autopilot and the autothrottles, as well as for entering speed, altitude, and other reference values into the autopilot. Below the flight control panel, in the center of the cockpit, extended a large, flat pedestal with three

computers, two stout-looking throttles, and an array of radios, switches, and levers. To the right of the pedestal sat the copilot, and to the left, the captain. They were each busy in their seats, preparing for the next flight. Behind them, on blue seats fixed to the cockpit wall, sat two observers—also pilots. The captain straightened his headset and began talking over the intercom, dictating the aircraft's configurations, modes, states, and performance numbers.

This was a test flight.

The aircraft was new; its maiden flight only eight months before. It was undergoing the rigor of repeated flight tests. Every component of the aircraft, from the wheels to the engines, all the way to the avionics suite, was tested in a multitude of conditions. By now 360 flight hours had been accumulated by various test pilots, not counting the hundreds of hours spent in the aircraft's flight simulator.

The captain, who was also the chief test pilot, grasped a small wheel-like crank in his left hand, looked back through his side window, and began making a tight U-turn. As the 200-foot-long aircraft finished the turn and moved slightly forward, the white, crosswalk-looking stripes, indicating the beginning of the runway, stretched on either side of the plane. As the aircraft came to a full stop with its main wheels resting heavily on the warm runway, the two pilots sank into their seats, resting from the just-completed test flight.

But within moments, they began preparing the aircraft for its second flight. They first entered the runway information and flight profile into the flight management computer. Meanwhile they were also monitoring their systems displays and checking the temperature inside the tires, waiting for them to cool from the just-completed landing, and reviewing the status of the hydraulic and electrical systems. As they were busy entering data into the computers and making sure the engines' temperature and other parameters were back to normal, the sun was already marching toward the southwest. It was after 5 P.M., and the workday was coming to an end.

The Last Flight

This last test flight was the culmination of a long and tiring day, especially for the captain: In the morning he met with representatives from a client airline and flew with them in another aircraft for an hour and a half, demonstrating the qualities of that aircraft. Afterwards he had jumped into the simulator of yet another new aircraft, testing and evaluating the aircraft's performance. Then he had a long lunch with the clients from the morning, followed by an interview with a Japanese TV crew. By mid-afternoon he was back on the flight line, where he met the copilot and flight-test engineer in preparation for these

two test flights, which began soon after 3 P.M. It was a busy and intense day, but just part of that unmistakable buzz that no visitor to a flight-test facility can ignore. The exhilaration associated with creating something new and a certain presence of danger is always in the air. Flying new aircraft and subjecting new designs to tests and maneuvers is a thrill, not unlike the excitement sustained on a movie set, in the sports arena, on a dance floor, or in any other highly creative human endeavor. Being part of it, participating in its tempo, is an engulfing lore that makes one go beyond the limits.

The flight-test engineer, from his console in the back of the aircraft, finished all his checks. He made sure that the autopilot was configured properly for the impending test. This flight was part of a required test to certify the autopilot and the engines. The test had been performed several times both in the simulator and also in the previous flight. This flight was going to be a routine flight—a repeat of what was already done. They were confident and relaxed. But the captain was tired from the long and eventful day, and as he looked to his right, he suggested that the copilot make this takeoff and fly the aircraft. The captain quickly briefed the impromptu change through the intercom. All of the test activities, from engaging the autopilot to shutting down systems, were done in coordination with the flight-test engineer who was sitting in the empty cabin. It was important for the captain to keep the flight-test engineer “in the loop” with respect to all activities that were taking place in the cockpit.

The control tower announced that wind was from the northeast (040 degrees), at 3-8 knots. The copilot mentally rehearsed the takeoff and the test sequence. They would be rolling fast down the runway, and then he would pitch the nose up and make a quick and aggressive takeoff. Seconds later, the captain would engage the autopilot—and then simulate an engine failure by reducing power on the left engine to imitate a condition in which the engine has quit on takeoff. Immediately afterwards, the captain was to shut off the left hydraulic system.

The aircraft was designed to perform adequately and climb safely with just one engine. And this capability had already been tested and demonstrated over and over during the preceding eight months. This time around, the engineers had suggested a small modification to the control logic of the autopilot. And part of this test flight was to evaluate the autopilot modification while the aircraft was flying on one engine only.

Testing and retesting the consequences of disturbance events such as engine failure on takeoff is one of the fundamental purposes of test flights. We never know when an engine will fail during a normal flight, with many passengers onboard—all we know, based on historical data, is that it might. Test flights are designed to ensure the airplane will survive the most extreme conditions it is likely to encounter. And therefore it is important to test and retest an aircraft in all kinds of flight conditions, well before the aircraft is fully

operational, in order to understand how the aircraft reacts under abnormal situations.

Test pilots are well trained for such meticulous testing. The captain was a graduate of a prestigious test-flight school, with considerable engineering experience and almost 8,000 hours of flying and test flights to his credit. He had flown 123 hours on this specific aircraft to date. The flight-test engineer, supervising all activities from his console in the back of the aircraft, had logged more than 6,000 hours as an engineer on such test flights. The copilot had more than 9,500 flight hours, and now he was ready to fly the aircraft and repeat the same test that they had just completed. During the previous flight, the autopilot modifications worked so well that they were too subtle to be even noticed. This flight was supposed to be no different.

The copilot was shifting in his seat, positioning his torso for the takeoff and placing his right hand on a small stick on his side. Instead of the traditional yoke-looking control wheel, this aircraft was equipped with an elegant side-stick controller. The copilot was going to be making a manual takeoff, pulling on the side stick to pitch the aircraft into the air. Shortly after becoming airborne, the captain was going to engage the autopilot—and from there onward it was going to be an automatic flight. The autopilot they were using was part of a sophisticated flight control system that included the autothrottles, flight management computers, and a variety of flight protection modes.

Envelope Protection

Figure 15.1 is a partial description of the autopilot and speed modes of this specific aircraft. In this autopilot, the **ALTITUDE CAPTURE** and **ALTITUDE HOLD** modes for transitioning from climb to level flight are the same as we have seen earlier in chapter 14. However, note that there are additional modes in the periphery. These are protection modes that engage automatically when the aircraft is in trouble, namely when the flight profile is unsafe and the aircraft is nearing a stall. Here's how the protection modes work: An onboard system senses and monitors the flight profile, constantly checking whether the aircraft is about to transgress beyond the normal and safe operating envelope. If the aircraft is coming close to a stall, a transition to the protection mode takes place automatically.

There are two protection modes in figure 15.1. The first one, **ALPHA PROTECTION MODE**, commands the aircraft to lower the nose to avoid a stall. However, just lowering the nose and thereby reducing the pitch attitude of the aircraft is necessary but not always sufficient to avoid a stall. When an aircraft is about to stall, it is usually flying at a low airspeed. Therefore, an additional protection mode, **THRUST PROTECTION**, is there to command full power from the engines, so as to increase speed and avoid a stall.

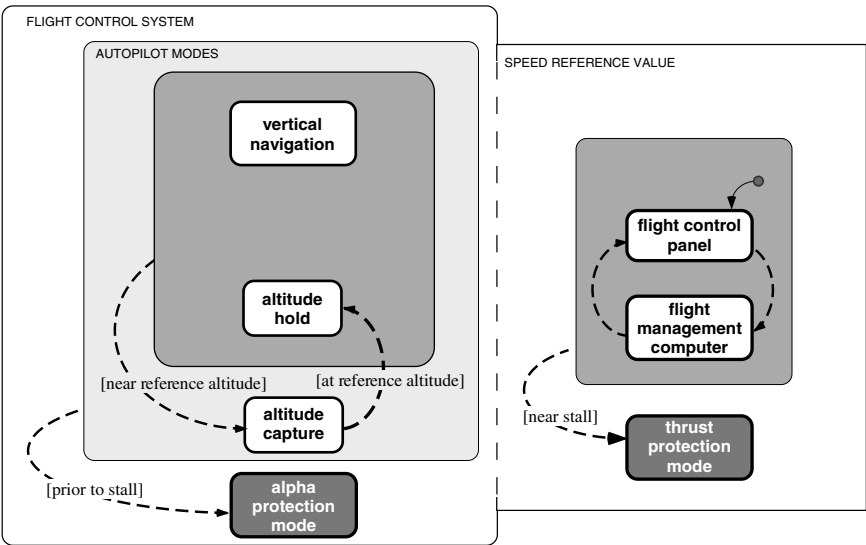


Figure 15.1. Autopilot modes and envelope protection.

As you can imagine, such *envelope protection* systems and their modes are extremely important in automated systems. We now find them in safety-critical systems such as medical equipment and aircraft, and they are also slowly migrating to automobiles. In any event, it is important to make a distinction between two categories of envelope protection systems: one, in which the envelope protection system protects against malfunctions or poor performance strictly on the part of the machine (such as in the case of the **RETRY** mode in the blood-pressure machine); the second category includes systems that override the user's actions in order to prevent an adverse effect. One such example is the antilock braking system (ABS), found in many new cars, that overrides the driver's braking pressure and prevents the wheels from skidding. Such systems improve on human abilities (even professional race drivers can't stop as quickly as an average driver with an ABS-equipped car) and prevent user errors and miscalculations during panic stops.

Envelope protection systems such as the antilock braking system—which is concerned with a single aspect of driving, namely braking and the prevention of skidding—have a very specific protection requirement. However, more complex envelope protection systems found onboard aircraft have to contend with multiple aspects of the flight such as preventing stall, ensuring that the pilot does not perform maneuvers that may cause structural damage to the aircraft, and preventing the aircraft from hitting the ground or other objects. Since there are multiple protection requirements (some of which interact in

non-trivial ways), and since there may be more than one way to protect the aircraft, taking the optimal protection action is a complex matter. Furthermore, making sure that the automatic actions that the envelope protection takes are *always* correct, given the context of the situation, is a major challenge facing the designers of such systems, as well as their users. The subject of automatic envelope protection systems that take authority away from the user has been a topic of hot debate, some of it quite emotional (especially in the aviation industry), and we shall talk more about this issue later.

Takeoff

From his observation platform towering high above the airport, the air traffic controller assigned to monitor and supervise the test flight had a magnificent view of the airport, adjacent farmlands, and the nearby city. A long black runway stretched below his feet. At the far end of the runway, the white aircraft was poised for takeoff. Turning his gaze back into his air traffic console, he made all necessary preparations to clear the aircraft for takeoff. The flight profile was going to be just like the previous one—a short takeoff run and a gradual climb to 10,000 feet, several slow maneuvers at altitude, and then a wide descending turn back to landing. The whole affair should take less than an hour. The air traffic controller checked the flight information of other aircraft and made a quick scan on the radar, making sure that there were no other aircraft in the way. He glanced at his copy of the flight-test order and visualized in his mind the takeoff and the straight climb to 10,000 feet. He looked up toward the white aircraft and quickly shifted his gaze toward the end of the runway and up to the open skies. His right hand clutched a little black box that he held near his belt and his finger pressed on its little “push to talk” button: “Cleared for takeoff.”

As the air traffic controller’s instructions rang in their ears, the copilot requested a confirmation from the captain as to which takeoff power setting to use. The captain responded, and the copilot repeated to himself: “takeoff thrust.” The copilot gently moved the two throttle levers forward, and the engines reacted with a long and high-octave whining. Within two seconds, the two big engines pulled the aircraft out of its parked spot. The wheels started rolling as the two large turbofans pulled all 325,605 pounds of metal, fuel, avionics, and crew forward. The aircraft was moving, its entire weight on the wheels, shaking slightly as it encountered small bumps on the runway; the wings drooped with heaviness and the tips were fluttering up and down in response. Within seconds, the speed was up to 50 knots, and increasing. And now the engines were pulling the aircraft and thrusting the long body forward.

The aircraft accelerated with every passing second and the ride became smoother as some of the aircraft’s weight was gradually transferred to the

wings. “80 knots,” called the captain. As the speed increased further, the long wings curved upward, and the wingtips rose in anticipation. At 100 knots the aircraft was ready to soar, but the copilot was pushing his side stick forward and holding the nose down; 120 knots, and he had to flex his hand muscle to prevent the aircraft from escaping free of the ground.

“Rotate,” commanded the captain at 132 knots, and the copilot pulled full aft on his side stick controller. The nose of the aircraft rebounded into the sky, and only the main wheels were still in contact with the ground. Within a second the nose was passing 6 degrees above the horizon and pitching up further. At that moment, the wings gave so much lift that the heavy aircraft had no choice but to let go its grip of the runway. The aircraft was airborne. Free of the ground, it began pitching up to 12 degrees, 14 degrees, and then 18 degrees. They were elevating into the sky at a high rate of climb.

As the long white aircraft soared to the sky, invisible contrails of data poured into the ground control center. Every word that was uttered in the cockpit was recorded. Every sensor in the aircraft—from those detecting airspeed, pitch attitude, and altitude, to those measuring engine thrust and hydraulic pressure—was recording information. The autopilot and autothrottles were not just recording and sending information about their active mode, but also about their internal states and parameters, and every reference value entered by the crew. Every two seconds, this telemetry information was batched and transmitted through a high-speed communication link to the ground control center. Most of the data was stored for further analysis; some of the data was displayed on computer screens inside the ground control center.

Four seconds after takeoff, the aircraft’s pitch attitude was already 20 degrees above the horizon, which is higher than the 15–17 degrees that we all commonly experience as passengers. The airspeed reached 155 knots, but then the high pitch attitude caused the aircraft’s speed to slowly decay.

But the aircraft was still full of thrust and uplifting energy when the captain called “Positive rate of climb,” and the copilot responded, “Gear up.” The captain instinctively leaned toward the instrument panel. The long day had exerted a toll on him, but everything here was familiar and he proceeded with business-like efficiency. He reached forward with his right hand and raised the little gear handle. “Clack,” came the metallic response as the handle hit the stopper, and then the landing gear began its familiar cycle, retracting into the belly of the aircraft. The captain moved his hand upward toward the long flight control panel. There, he went straight for the autopilot engage button and pushed it in. Meanwhile the aircraft’s nose was coming to a high 23-degree pitch attitude, and the copilot, noticing the unusual attitude and the gradual loss of speed, began manually pushing down on his side stick and lowering the nose.

But the autopilot did not engage.

The captain pushed on the button again. No engagement. After several successive tries at the button, each time harder, the autopilot finally engaged and he could see the proof of this on his display. Relieved, the captain's mind raced on and was already preoccupied with the next task in the flight test order. What he did not see was that the aircraft's speed was slowly coming down to 150 knots, as the engines, although in full forward thrust, could not keep up with the high pitch attitude that had already attained 24 degrees and was still going strong.

The autopilot, for its part, was still resisting engagement in a passive way. Although engaged, it did not activate any mode, nor did it take control of the aircraft. The captain, however, did not notice all of this as he was already preparing for the next task—reducing thrust to idle on the left engine to simulate an engine failure. The pitch attitude continued to inch up as the captain's hand flew toward the throttle quadrant. He grabbed the left throttle with his hand and gently brought it back; making sure it was firmly against the idle stop. He then quickly went on to his third task: shutting down the left hydraulic system.

As the captain's eyes shifted to the overhead panel and his hands reached up for the hydraulic system controls, the autopilot mode indications went blank. This was by design. The systems logic was that if the aircraft's attitude was beyond 25 degrees, certainly an abnormal situation, the pilot should not be bothered with superfluous mode information. The display now showed only the airspeed, heading, altitude, vertical speed, and the pitch attitude, which had already reached 28 degrees nose up when the left engine ceased to produce thrust. The aircraft was now flying on one engine, 500 feet above the ground, speed decreasing below 145 knots, and its nose pointing 29 degrees up into the sky. The aircraft was heading towards an undesirable, abnormal, and potentially dire situation.

Autopilot Taking Control

At that moment, the autopilot activated and assumed control and then immediately transitioned to **ALTITUDE CAPTURE** mode. But the displays were omitting this information, and the pilots had no way of knowing which mode the autopilot was in. A second later, the **ALTITUDE CAPTURE** mode began guiding the aircraft along a parabolic profile to attain 2,000 feet, which was the reference altitude setting on the flight control panel.

Initially, the autopilot gave nose-down commands, and the pitch attitude came down to 28, then 27, and then 25 degrees. It seemed as if the autopilot was finally taking command and assuredly returning the aircraft to a normal and safer climb profile. But a second later, the autopilot reversed its commands; instead of continuing to lower the nose, it gave nose-up commands.

The aircraft pitched upward, and began racing into the sky again—its nose at 26 degrees, 27, 29—as if the autopilot was desperate to climb even faster, in spite of the fact that the aircraft's thrust had been cut in half, and its left wing had begun to drop.

"Pump fault," said the captain in a casual voice, confirming the action he had taken to disable the hydraulic system in keeping with the test-flight order. With his eyes and attention focused on the hydraulic panel up in the cockpit's ceiling, he was probably oblivious to what was going on below, because at that moment the cockpit was pointing 30 degrees up in the air and the autopilot was progressively commanding higher and higher pitch attitudes.

But the laws of physics could not be fooled for long. As the aircraft began to lose its momentum and the pitch attitude only increased with fervor, the aircraft's speed began to spiral downward to 129 knots, then 120 knots. The aircraft was now losing speed at an alarming rate, passing through 118 knots, which is the lowest speed that the aircraft can still be fully maneuvered. In two seconds it was down to 113. The situation, like the speed, was deteriorating rapidly. Yet the autopilot kept coercing the aircraft on a fanatical and unattainable march into the sky. When the captain finished with the hydraulic system and his eyes and hand came down from the overhead panel, the blue horizon line was hidden below the nose, which was suspended 30 degrees up in mid-air.

"What happened?" he said in surprise.

The Captain's Takeover

It took the captain three seconds to look at the instruments and realize the severity of the situation. He immediately disconnected the autopilot and assumed manual control. He now had a crippled aircraft in his hands and he felt through his body and feet the vulnerability of the aircraft. The nose was pointing 31 degrees into the sky, speed was at a meager 100 knots, the left engine at idle and the right engine producing maximum thrust. The asymmetric thrust between the left and right engines was beginning to exert a corkscrew effect, pushing and rolling the aircraft to the left. But worst of all, the aircraft was just 1,400 feet above the ground. "Speed, Speed, Speed" chirped a metallic voice from the envelope protections system as the left wing dropped down further. The captain was fully cognizant that the aircraft was no longer fully maneuverable, and he knew he had to act fast in order to recover. But he wasn't the only one: the automatic envelope protection system also sensed that something was wrong, and immediately transitioned to ALPHA PROTECTION mode, which, in turn, commanded a nose-down attitude.

But the captain was faster.

He was already pushing the nose down, all the way, in an effort to regain speed and maneuverability. He also initiated a right turn, trying to hoist up the

left wing. Sensing the heaviness of the wing, he stepped hard on the right rudder to prevent the aircraft from entering a left turn. Two seconds later, realizing that nothing was going to help—he pulled back on the right throttle.

The right engine was the only engine producing full thrust and keeping the aircraft flying forward. But as startling as it may sound (because now the aircraft became a glider), this was the only choice available to the captain, and he acted decisively. He had to bring the left wing back, and reducing thrust on the good engine was the only way to even out the corkscrewing effect that was pushing the aircraft to the left and threatening a downward spiral. But it was too late, because the left wing already began to buffet from the disruption of the airflow.

A second later another protection system woke up: the **AUTOMATIC THRUST PROTECTION** mode engaged, and commanded both throttles to full power, in an attempt to pull the aircraft away from the stall. But since the captain already pulled back on the right engine and both engines were at idle stop, the activation of this automatic protection mode was canceled. The aircraft was already in dire straits, and there were no more protections in store.

The pitch attitude was so high, and the speed so low, that the long and broad wing could no longer sustain itself and began sinking rapidly as the speed passed through 77 knots. The aircraft's body was shaking from turbulent confusion and the left wing dropped violently, throwing the aircraft into a brutal left turn, while a cacophony of computer-generated warnings continued sounding in the cockpit. And although the pitch attitude was steadily coming down to 0 as the captain was pushing the nose down, the aircraft wings reached 90 degrees vertical and continued to a 110-degree roll, leaving the crew hanging upside down, held to their seats only with their shoulder harnesses. But the captain kept on pushing the nose down, and the long aircraft began responding rapidly; the pitch attitude dropped to 20 degrees down as the view of the black runway and surrounding countryside rushed to fill the cockpit window. The speed began to increase: 90 knots and then 110. Slowly at first, and then with increasing vigor, the left wing began to recover and rose up to reverse the roll. A second later, the speed was 125 knots and increasing, the left wing returning back to level, and the aircraft was maneuverable again.

But the altitude was only 600 feet.

The nose was pointing down 35 degrees while the captain reversed his actions and began pulling hard to raise the nose and avoid the ground that was racing toward him at 7,500 feet per minute. The nose cooperated and began pulling out, trying to prevent the dark runway from taking them in. But the large and heavy airliner—more than 200 feet in length—could not muster the excruciating maneuver needed to escape the ground. Four seconds later, with the captain and copilot pulling as hard as they could on the controls, and every

rivet, beam, and metal straining to prevent the worst—the white aircraft slammed into the ground. Within seconds it was engulfed in flames.

There were no survivors.

Why Did it Happen?

The entire flight lasted only 36 seconds and the aircraft crashed near the end of the runway from which they took off. Figure 15.2 is the flight profile. Like any accident, this tragic crash is a sequence of many abnormal events, each one not dominant enough to cause a catastrophe, but their combined effect did. We will start with the machine and its behavior, and then proceed to consider the pilot interaction with the automation.

The autopilot engaged six seconds after takeoff following repeated attempts by the captain. The altitude was 500 feet and the aircraft was catapulting into the sky at 6,000 feet per minute. As soon as it activated, the autopilot switched to **ALTITUDE CAPTURE** mode and began the curved maneuver to capture 2,000 feet.

Why 2,000 feet when the flight-test order called for a straight climb to 10,000 feet?

The post-accident investigation found that the autopilot's (reference) altitude was indeed set to 2,000 feet. One supposition for this discrepancy is that the captain and copilot forgot to reset the altitude to 10,000 feet; another is that they did intend to level off at 2,000, but coordinated this out-of-the-ordinary change to the test order using only hand signals (the cockpit voice recorder did not contain any mention of this 2,000 feet level-off modification). Since there was no record of any intercom contact with the flight-test engineer, who was sitting in the cabin away from the cockpit, regarding this change, nor any communication with the air traffic controller about such a major departure from the flight test, we will continue our discussion assuming that the crew *forgot* to set 10,000 feet.

If that's the case, where did the 2,000 feet come from? The only plausible explanation is that it was left over from the previous flight. And indeed, it turns out that the last altitude that the crew had entered into the flight control panel during the previous flight was 2,000 feet. For some reason, perhaps from being tired or for other reasons, they forgot to reset the altitude to 10,000 feet after landing and failed to notice this discrepancy during the preparation for the second takeoff.

The Autopilot

The captain tried several times to engage the autopilot. It did not engage and the captain had to push several times on the button to make it happen. The

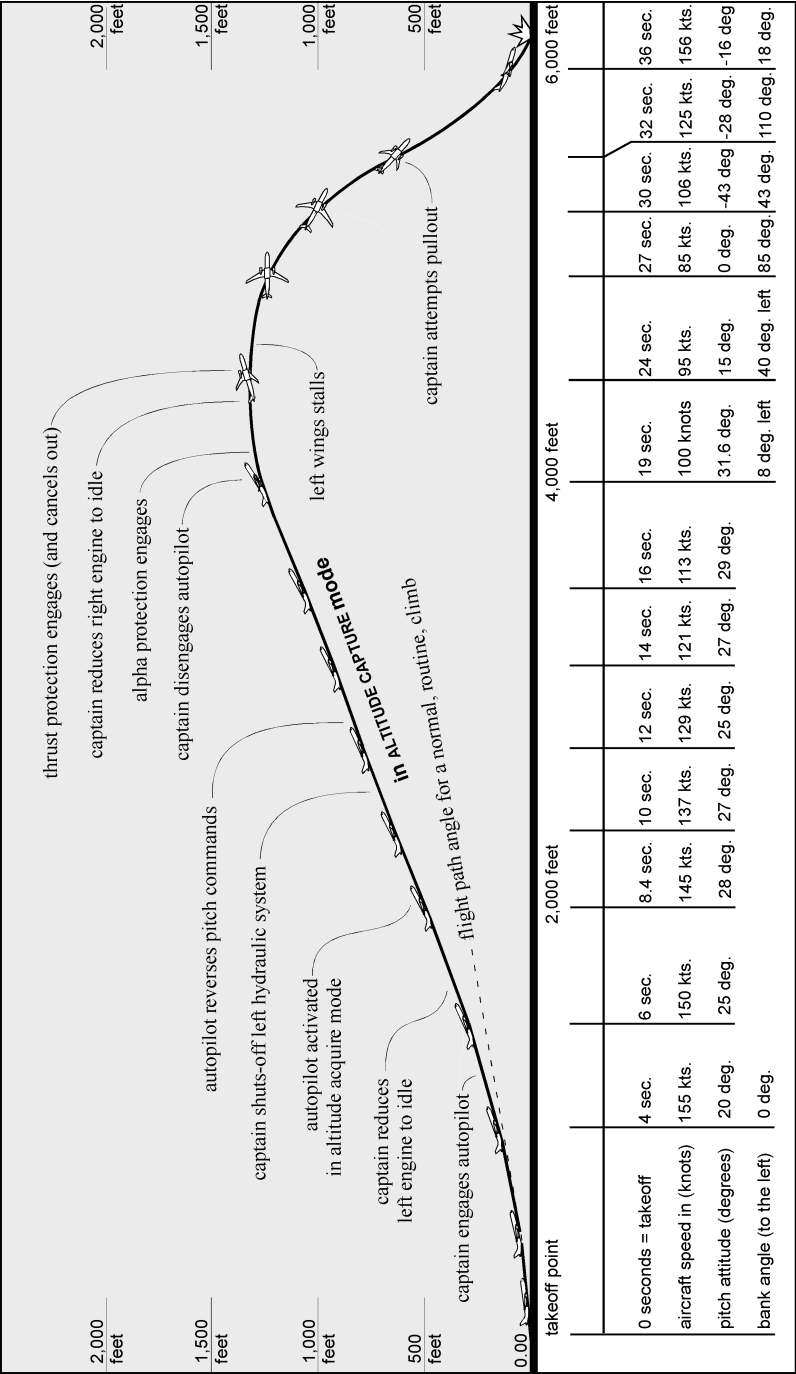


Figure 15.2. Reconstruction of the flight profile.

culprit was not the button, however. It was actually the autopilot's internal logic. When the captain tried to engage the autopilot, the copilot was exerting a slight nose-down pressure on his control in order to counter the high (25-degree) pitch attitude. There is a built-in logic in the autopilot that resists engagement of the autopilot (and also delays activation) when the pilot is making manual inputs. When the autopilot finally activated 2 seconds later, the pitch attitude was 29 degrees.

After the autopilot took control, it initially lowered the high nose-up attitude back to 25 degrees. And then 4 seconds later the autopilot reversed itself and progressively increased the pitch attitude—attaining almost 32 degrees by the time the captain intervened. If the captain had not disconnected the autopilot and assumed control, the autopilot, left on its own, would have probably continued increasing the pitch attitude beyond 32 degrees.

Why?

Post-accident analysis of the autopilot logic revealed a design flaw, which although it had existed for a long time, was neither detected nor encountered during software testing, simulator tests, or flight tests. This flaw had to do with control authority. When the autopilot transitions to **ALTITUDE CAPTURE** mode, it immediately calculates the rate of climb it needs to follow in order to close on and reach 2,000 feet. At the time the calculation was made, there were two strong engines producing full thrust.

And this is the underpinning of the entire accident, because shortly after the profile calculation was done by the autopilot, the captain reduced power on the left engine (to simulate an engine failure). At that point, the aircraft had only one engine producing thrust. The right engine could not produce enough thrust to keep the aircraft flying on the steep profile. But the autopilot did not care; it wanted to close the gap to 2,000 as fast as it could. The autopilot was not designed to progressively calculate the altitude capture profile; rather it was designed to follow the rate of climb that it computed initially. And follow this rate blindly, it did. The aircraft, nonetheless, could no longer fly that steep profile because its left engine was idle and not producing thrust. As a consequence, the aircraft began slowing down, falling off and deviating from the climb profile (see figure 15.3).

But as soon as the autopilot detected the deviation, it responded immediately by pitching up and increasing the aircraft's attitude in an attempt to climb back again. Yet the increased pitch attitude further reduced the aircraft's speed and caused an even greater deviation. Then the autopilot commanded an even higher pitch attitude. It was like a snowball effect—because every increase in pitch brought about an even greater deviation. Dumb and dutiful, the autopilot progressively commanded higher and higher pitch attitudes, well beyond the normal values, in a futile and ignorant attempt to make the aircraft fly a flight profile that had become impossible to attain. The autopilot was guiding the aircraft into a stall.

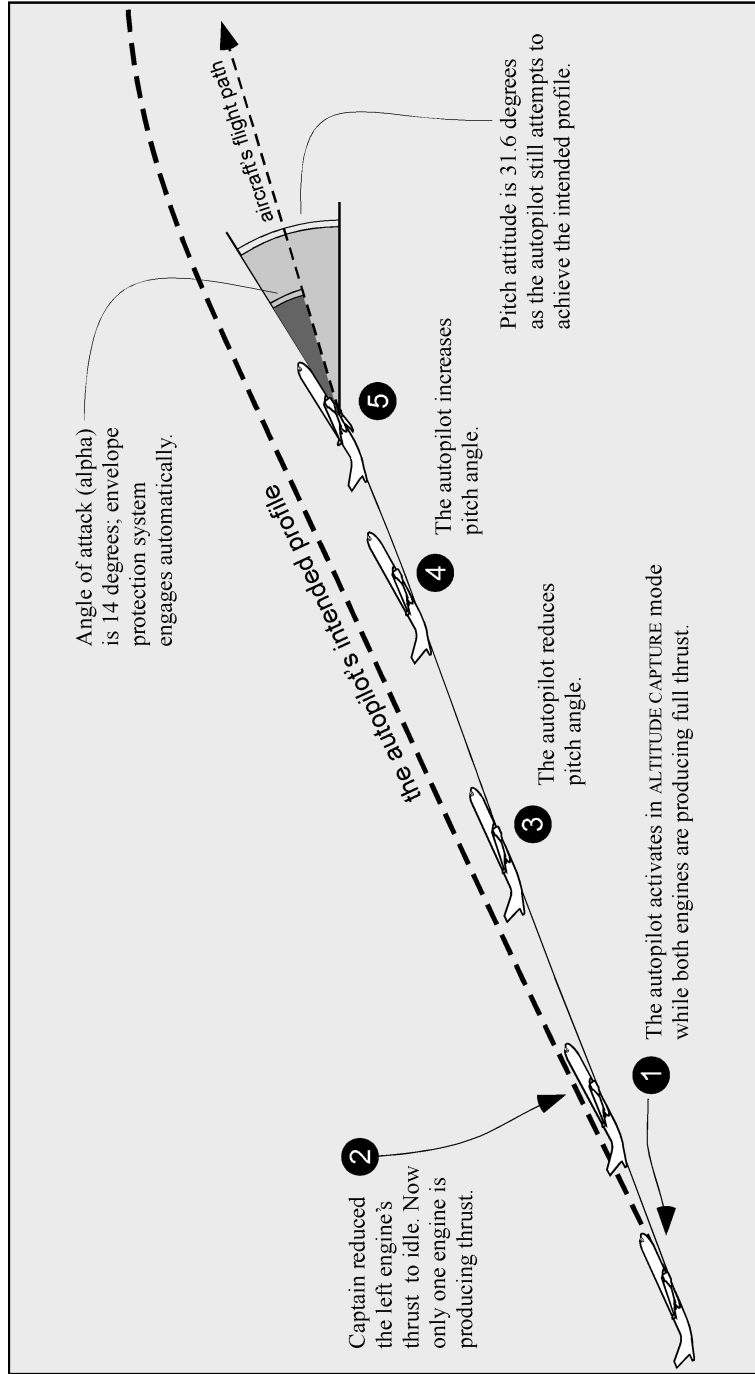


Figure 15.3. Aircraft's profile (not to scale).

Limited Protection

The flight control system, however, was designed with envelope protection modes to prevent the aircraft from entering a stall. To understand why the protection modes did not take effect, we need to go a bit deeper into what these systems can do and what they can't do.

Envelope protection systems provide a cocoon-like shell in which the autopilot (or the pilot) can do whatever he wishes to do—but once the system is about to go beyond the safety of the cocoon, the protection modes kick in and attempt to bring the aircraft back to safety. What this implies is that the envelope protection system splits the entire operational space (which is defined in terms of speed, pitch attitude, and other parameters)—into two separate regions: a normal operational region and an unsafe operational region. Figure 15.4(a) is an abstract description of the operating region and its division. The envelope protection system constantly monitors the location of the aircraft with respect to the unsafe region. The goal of the envelope protection system is to detect a transgression from the normal operational region into the unsafe region. And when that happens, the system takes control and tries to push the aircraft back into the normal and safe operational region.

Because envelope protection systems are in place to prevent the system from going unsafe, we want them to kick in *just before* the system crosses the boundary into the unsafe region. This is because it may take the system some time to reverse its actions and return to the normal region. This implies that there is an additional region, which I shall call the *buffer zone*. When the aircraft enters the buffer zone, the envelope protection system takes over and guides the system back before it reaches the unsafe region. The size of the buffer zone, which is always embedded in the normal region, depends on the dynamics of the aircraft.

Figure 15.4(b) is a graphic depiction of how the envelope protection system works. Point “a” corresponds to the aircraft operating in the normal region, and “b” is where the envelope protection system engages. Point “c” shows that the aircraft is already inside the buffer zone, but just skirting the unsafe region and is safely ushered out. In “d” the aircraft has returned to the normal region.

Envelope protection systems are critical for the sophisticated automation of today and the future because they provide an additional level of safety. Envelope protection systems bestow on us confidence that the system will not become unsafe and can always recover. Yet envelope protection systems are very difficult to design and should not be accepted naïvely by users as some kind of a fail-safe. There are several major issues that make the design of these Golems of our time rather complex: one, determining which cues truly and reliably indicate that the system is indeed crossing into an unsafe situation; two, establishing an appropriate buffer zone between the safe and unsafe

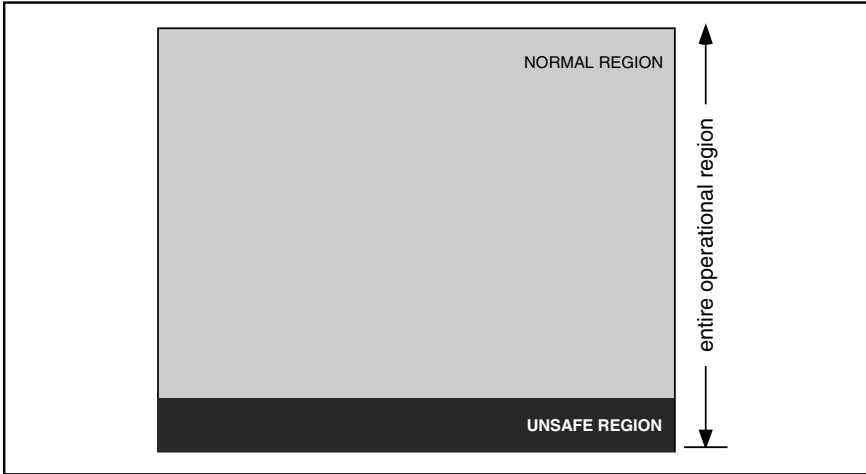


Figure 15.4 (a). Normal and unsafe operational regions.

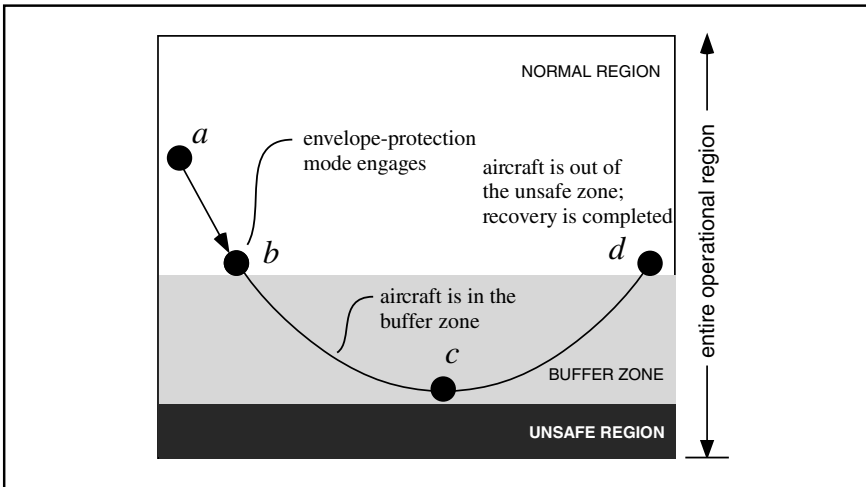


Figure 15.4 (b). Buffer zone and recovery.

regions of the operational space, which, as we mentioned earlier, has to do with the dynamics of the system. For example, for a really fast aircraft, the buffer zone must be very large—because it will take longer for the envelope protection system to bring the aircraft back to safety.

With this in mind, let us return to the particulars of the accident. The envelope protection system onboard the aircraft was designed to protect the aircraft from a stall. It does this by sensing the aircraft's *angle of attack*, which, generally speaking, is the angle between the nose-up (pitch attitude) and the aircraft's flight path (see figure 15.3). This angle of attack, termed *alpha* in

aeronautical engineering lingo (and hence **ALPHA PROTECTION** mode), is quite indicative of stall. When this angle gets too high, stall occurs.

Nineteen seconds into the flight, when the pitch attitude reached 31.6 degrees and was increasing, the *angle of attack* exceeded 14 degrees. The guard on the mode transition to **ALPHA PROTECTION** mode became true and the transition took place (see figure 15.1). The **ALPHA PROTECTION** mode commanded the aircraft to lower the nose. Five seconds later, the second envelope protection mode engaged. The **THRUST PROTECTION** mode commanded full thrust on both engines, in an attempt to move the aircraft out of the unsafe (stall) region.

The captain, however, was faster than both protection systems: he had already commanded full nose-down attitude (in the same direction as the **ALPHA PROTECTION** mode) and then pulled back the right throttle, which canceled the full thrust that the **THRUST PROTECTION** mode was commanding. Even if the captain had not intervened manually, it is unlikely that the envelope protection system would have made a successful recovery. Why? Because there was simply not enough altitude for the airplane to recover. And here is a subtlety about the envelope protection system that is important to understand: the system sensed only what the aircraft was doing—it was not designed to monitor and take into account the aircraft's altitude.

At the time that the envelope protection system engaged, the altitude was only 1,300 feet. The aircraft simply couldn't switch from the situation in which it was about to stall to a full recovery, given such limited altitude. Had this problem occurred at 10,000 feet, the outcome might have been very different.

This issue of the time (and resulting loss of altitude) it takes the aircraft to recover and return to the normal operating region is an important consideration in the design of the buffer zone. Naturally, for a 200-foot-long, 325,000-pound aircraft, it takes quite some time to recover. The transition to the envelope protection mode occurred when the aircraft's angle of attack reached 14 degrees—but this was too late considering the aircraft's altitude (1,300 feet). For the aircraft to recover, given its low altitude, the transition to **ALPHA PROTECTION** mode should have occurred much earlier.

Note also that the quality and integrity of envelope protection systems depend on many factors, including (1) the internal events (e.g., speed, angle of attack, engine thrust) that are being sensed and used as conditions for transitioning into the protection mode; (2) the status of the environment (altitude, distance to obstacles such as buildings and mountains, and so on); and (3) the possible disturbances (in this case engine failure, loss of all hydraulic systems) and the resulting limits on aircraft performance. Therefore, the size of the operating envelope constantly changes during a flight. With it, there are constant changes to the size and shape of the buffer zone and the unsafe region.

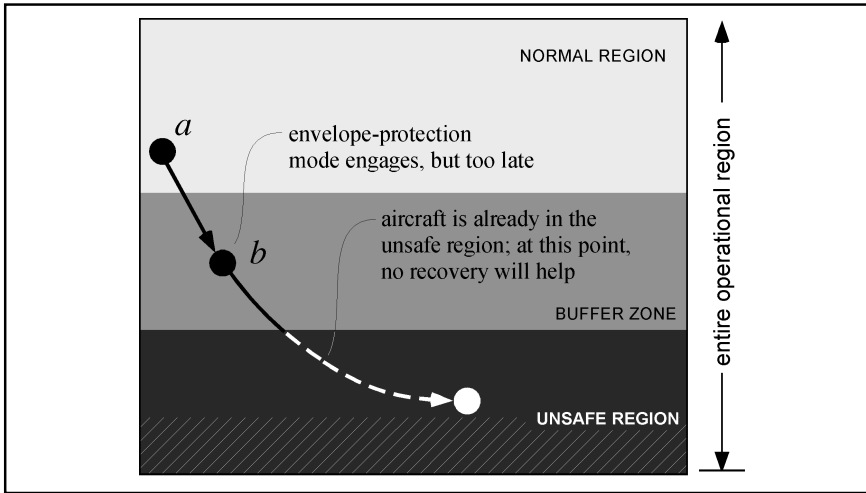


Figure 15.5. Shrinking of the normal and safe operational region (and expansion of the unsafe region) when altitude is considered.

Figure 15.5 shows the same recovery maneuver as seen in the previous figure. But here we take into account the aircraft altitude. Accounting for this factor shrinks the size of the operational region. The unsafe zone is bumped up, as is the buffer zone, demonstrating that for the recovery to succeed, the engagement of the envelope protection maneuver must begin earlier.

Erroneous Trim Setting

But even before the captain intervened and the envelope protection engaged in an attempt to save the situation, the aircraft, under the copilot's manual control, reached a 25-degree nose-up attitude. For comparison, the highest pitch attitude reached during the previous flight was only 14 degrees; during a routine takeoff on a commercial flight with passengers onboard it hardly ever exceeds 15–17 degrees. So the nose-up attitude indeed was unusual, and there was no lack of indications in the cockpit about this high-pitch attitude. Therefore, one of the immediate questions that arises is why did the copilot allow the pitch attitude to reach such a high angle in the first place?

There isn't a definitive answer to this question, but there are several explanatory factors in the way the aircraft was configured for takeoff: The longitudinal (pitch up or down) stability of the aircraft can be adjusted by setting the aircraft's trim (the pilots set the trim by moving the aircraft's horizontal stabilizer up and down). Trim adjustment is used to counter situations where the aircraft is loaded such that most of the weight is behind the wings, which will cause the nose to rise during flight. Figure 15.6 shows

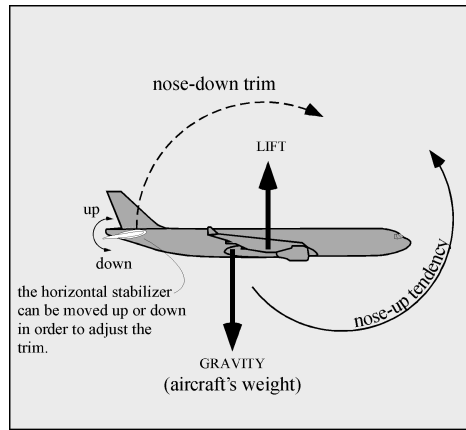


Figure 15.6. The aircraft's center-of-gravity is behind the center-of-lift. The resulting momentum causes a nose-up tendency. The pilots counter this effect by moving the aircraft's elevator (located in the tail of the aircraft). In the figure, the elevator is adjusted for a nose down effect.

that when the aircraft's center of gravity is behind the aircraft's center of lift, the pilots must adjust the aircraft's trim for nose-down (to counter the aircraft's tendency to lift up the nose).

During the test flight accident, the aircraft's center of gravity was behind the aircraft's center of lift, but the trim setting was set for a *nose-up* effect (2.2 degrees). It is not clear why the crew set the trim for a nose-up effect (the last setting used during the previous landing was 4.0 nose-up). But at any event, it created an even stronger nose-up tendency during takeoff and climb. (On their first takeoff, by comparison, the trim was set near zero degrees.)

The combination of a high nose-up tendency and maximum takeoff thrust caused the aircraft to pitch up aggressively during the climb. The copilot could have manually pushed the nose down to stop the aircraft from pitching up and prevented the autopilot from progressively commanding such excessive pitch attitudes. It may have been his familiarity with and confidence in the aircraft's autopilot and protection systems that made him less concerned about the steep climb profile.

Trust in Automation

Then comes the question of why the copilot did not disengage the autopilot when it commanded even higher pitch attitudes. Perhaps knowing that all this was part of an autopilot test, the copilot felt that they could go beyond the normal flight envelope. After all, that's what test flights are for. And maybe the copilot believed they should explore how the autopilot would correct such high pitch attitudes, even though the flight-test order did not call for such a

demonstration. It is also possible that the copilot's failure to take action and disengage the autopilot may also have had something to do with a phenomenon called "trust in automation."

Most automated control systems are extremely reliable and safe. Over the course of human-automation interactions, just as in human relations, trust is built. Nevertheless, on rare occasions, automated systems fail. Over-trust in automation can bring users to the point of not recognizing the machine's inherent limitations. It can also lead to situations where the users fail to monitor and supervise the machine adequately. But worst of all, over-trust on automation can lull users into ignoring and dismissing cues that indicate the machine is failing to act as expected, and the situation is becoming progressively more dangerous. The sequence of events that led to the grounding of the *Royal Majesty* (chapter 8) is one sad example of this pervasive problem.

In aviation, over-trust on automation is a well-documented phenomenon. There were several aircraft accidents that involved inadequate monitoring of autopilots, and the National Transportation Safety Board (NTSB) began alerting the aviation industry to this problem as early as 1973, following the crash of an Eastern Airlines L-1011 aircraft in the Florida Everglades. (In that accident, the aircraft was on autopilot control, flying level at 2,000 feet and the entire flight crew was busy attending to a suspected landing gear malfunction. The autopilot, in ALTITUDE HOLD mode, disengaged unexpectedly. But since nobody was watching over the autopilot and monitoring the aircraft's flight path, the aircraft gradually descended into the ground.)

Several psychological experiments have shown that reliance on the decisions and actions of automated systems can make humans inattentive to information indicating that the automation is in trouble. Specifically, these experiments demonstrate that pilots have a strong tendency to trust automated systems, in spite of clear evidence from other cockpit displays that the automated system is misbehaving. In other words, pilots have a tendency, built on past experience, to *not* disengage the automation even when it goes wrong. Both the U.S. Federal Aviation Administration and several industry groups have publicly expressed concern about pilots' reluctance to disengage automated systems and take manual control of the aircraft. Further, although here we are discussing this issue in the context of flight crews and cockpit automation, anecdotal evidence indicates that the problem of over-trust is not unique to pilots. This issue is also a major concern in the design of futuristic automotive systems such as adaptive cruise controls and intelligent highway systems.

Crew Resource Management

When the aircraft was experiencing abnormal nose-high attitudes, the captain was busy with the flight-test tasks (turning the autopilot on, bringing the left

engine to idle, and working the hydraulic system). However, neither the copilot nor the flight-test engineer said anything to the captain about the increasingly deteriorating situation.

The concept of Crew Resource Management emerged in the early 1980s in commercial aviation as many began to realize the importance of crew coordination and its impact on safety. The value of this concept was reinforced by several accidents, in which critical information (such as fuel status) was not shared and communicated among crewmembers. Lack of crew coordination, in which pilots acted alone and did not communicate clearly with fellow crewmembers, led to wrong decisions and catastrophic outcomes. Following several such accidents in which the post-accident investigation clearly indicated that a breakdown in crew coordination was a dominant factor, most major airlines began providing Crew Resource Management techniques to their pilots.

Dialogue between the crewmembers before the last test flight revealed that the two pilots and the flight-test engineer had a busy schedule that day, and they were tired. It also seems that although the captain and the copilot switched their roles just before the flight, they did not fully discuss or review the assignment of tasks. In particular, the copilot may have believed that once his takeoff was completed, and the autopilot engaged, he (the copilot) was relieved from his monitoring task. This may have affected the copilot's reluctance to disengage the autopilot when it attained such a high pitch attitude and explain his lack of communication with the captain. The corresponding problem was that the captain, who was occupied with bringing down the left engine to idle thrust and configuring the hydraulic system on the cockpit ceiling, was relying on the copilot to monitor the aircraft and the autopilot's behavior. The flight-test engineer also failed to alert the captain to the increasingly dangerous pitch attitude and the deterioration in airspeed. His first warning that something was very wrong came three seconds after the captain had already disengaged the autopilot and was well into his recovery attempts.

In Conclusion

This accident revolved around two issues: the problem in the autopilot's ALTITUDE CAPTURE mode and the recovery efforts. There was a serious deficiency in the autopilot's design. The autopilot had the authority to apply extremely high pitch attitudes. In this particular accident, the autopilot used its control authority to guide the aircraft along a capture profile that was no longer attainable. Following the accident, an analysis was conducted on autopilots used on other manufacturers' aircraft. Almost all autopilots had the same

design problem, which rendered them vulnerable to the same kind of mishap. These findings prompted aircraft and avionics manufacturers to add software features to fix the problem—many of them opting for a software routine that limits the autopilot's pitch attitude in **ALTITUDE CAPTURE** mode.

Some engineers further argue that beyond the immediate solution that was implemented, a lack of integration between the autopilot and the autothrottle (that controls engine thrust) is the real culprit. Due to the piecemeal introduction of cockpit automation, the autopilot and the autothrottle are not fully integrated, leading to this and other problems that designers of autopilots, and ultimately pilots, must work around.

With respect to the recovery, there were two “agents” participating in these efforts: the captain, on the one hand, and the automatic envelope protection system on the other. After takeoff, the captain concentrated his efforts on the tasks prescribed in the flight-test order. His third task, shutdown of the left hydraulic systems, was completed 10 seconds into the flight when the speed was 135 knots and the pitch attitude was 27 degrees. Five seconds later he realized that something was abnormal and snapped, “What happened?” Three seconds later he disconnected the autopilot. Altogether, eight seconds elapsed between the captain's shutdown of the left hydraulic system and his disengagement of the autopilot.

One of the daunting questions that this accident raises is why it took so long for the captain to disengage the autopilot and begin the recovery. Given that the entire flight lasted about 36 seconds, 8 seconds constitutes almost a quarter of the time aloft. When we account for human reaction time (about 1.5-2 seconds in such situations), both for (1) recognizing and understanding the situation, and (2) disengaging the autopilot, it reduces this period to 4-5 seconds. It is unclear what went on during this period: Could it have been that the captain was preoccupied with yet another task, perhaps taking notes, showing something to the two pilot observers sitting behind him, or executing yet another task or procedure? Likewise, while he was occupied with executing the flight-test order, the aircraft's pitch attitude was progressively increasing well beyond normal pitch attitude. What kept the captain from sensing this acute nose-high attitude? Unfortunately, the accident report provides no answers to these perplexing questions.

What is obvious, however, is that the disappearance of the autopilot mode indicators prevented the captain from quickly recognizing and understanding the seriousness of the situation at hand. He could not tell what mode the aircraft was in, and that quandary may have contributed to the 4-5 second delay mentioned earlier. Yet when he disconnected the autopilot and began the recovery effort, the captain acted correctly and very quickly. Subsequent simulations have shown that if the captain had begun his recovery efforts just four seconds earlier, the same maneuvers that he so precisely executed would

have prevented the accident and saved the day. This accident clearly demonstrates how the decision to disengage or keep using an automated system becomes one of the most critical decisions that a user must make while interacting with a safety-critical system, especially during an emergency. But hindsight, as we all know, is always 20/20.

Finally, it is important to note the unique interplay between the envelope protection system's engagement and the captain's recovery attempts. It is precisely in these critical moments, in which every second counts, that we see the potential for dangerous situations in which both the machine and the human are making simultaneous, yet uncoordinated, efforts for recovery. In the test flight accident described above, while the **ALPHA-PROTECTION** mode was engaged and active and while the **THRUST PROTECTION** tried to engage, the captain was interacting with the aircraft's controls alongside with the automation. We have already seen how "competition" between pilot and autopilot can lead to disaster (in an earlier discussion in chapter 3). The possibility of dual authority and, more poignantly, the possible partition between authority and responsibility, creates a serious potential for operational failure. Responsibility should always be accompanied with suitable authority. Therefore, if indeed the envelope protection responsibility is to rest with the automation, the pilot cannot manipulate the controls at the same time (yet the pilot should be given the authority to *completely* override the automation and take *full* manual control). These issues are far from simple and immediately raise critical questions. What should the pilot do when the machine takes action? What if the pilot's instinctive responses are in contradiction to the machine's actions? And then there is the ultimate question: who can, and who should—especially in these split-second situations—override whom?

An Important Lesson

The classic story of a computer that overrides the users, locks them out, and takes violent action is that of HAL. In *2001: A Space Odyssey*, the super computer HAL-9000 appears to be sensitive, intelligent, and trustworthy. But then HAL begins to malfunction as the spaceship *Discovery* comes closer to its destination. Both ground control (on Earth) as well as the two (awakened) crew members realize that HAL is making errors. Ground control suggests temporarily disconnecting HAL, transferring control to another computer on Earth, and then repairing HAL's programs; the crew wants to override HAL and take manual control of the ship. But the computer is unable to accept the fact that its internal logic circuits are failing ("no 9000 computer has ever made a mistake or distorted information") nor can it accept a disconnect and an override; HAL wasn't pre-programmed for such a situation. The only thing HAL can do is to keep its pre-programmed task of continuing the mission at all

costs. HAL begins to kill the astronauts on board in an attempt to continue the mission on its own. It is this acute problem of a machine that is given considerable authority, yet has limited ability to understand a situation, which produces the drama of *2001: A Space Odyssey*.

For all of us to avoid the HALs of the future, it is important to address such problems of authority, responsibility, and interaction head-on. It is the role of regulatory agencies (such as the Federal Aviation Administration, Nuclear Regulatory Commission, and Food and Drug Administration in the United States) to provide guidelines, requirements, and criteria for safe and reliable automated systems. For example, in aviation there are certification committees comprised of engineers, test pilots, and scientists that are tasked with developing design criteria for automated control systems. But for these committees to be effective, design guidelines and criteria must be based not only on in-depth understanding of how these machines work, but also on how humans can and should interact with them.

Modern-day machines, computers, and automated systems cannot think. As compelling as it may appear to a user who is working for days on end with a flawless computer or automated system, a machine cannot “read” an unexpected situation or scrutinize it. The autopilot described in this chapter continued increasing pitch attitude beyond reasonable angles with complete disregard of the situation and the consequences. It did what it was pre-programmed to do.

At its very essence, a computer is a system of states and transitions in-between. The computer senses an event and switches from one state to another. The description of this (pre-programmed) behavior is what you have in every model in this book—from the on/off light switch in chapter 1 (figure 1.2) to the model of the autopilot and envelope protection in figure 15.1. The reason a machine can fly an aircraft and guide a spacecraft to other planets, lies in the sophistication of what was programmed into the machine; namely, its map of states and transitions. Modern-day machines can only follow that map—nothing more and nothing less.

The term “automation surprises” was coined by Nadine Sarter and David Woods in the early 1990s following their work on the human factors of automated cockpits. This work has received a lot of attention and helped designers and pilots to better understand the problems involved in operating complex automated systems. (See the chapter titled “Automation Surprises,” by David Woods, Nadine Sarter, and Charles Billings, in the *Handbook of Human Factors and Ergonomics*, published in 1997 by John Wiley.) Another related term that is sometimes used to describe such human-automation interaction problems is called “mode error.” The term was coined by Donald Norman in the early 80s to categorize a special type of human error and is described in *The Design of Everyday Things* (2002, Basic Books). The importance of feedback to the pilot about the state of the automation is stressed in a classic article by Donald Norman, published in 1990, titled “The ‘Problem’ with Automation: Inappropriate Feedback and Interaction, Not ‘Over-Automation’” (*Philosophical Transactions of the Royal Society of London*, B-327, 585-593).

There are several other (speed-related) problems with the interface to the autopilot described in this chapter. Look again at figure 14.6 and note that while the phone’s design has the automatic transition only in one direction (from OFF to ON), the speed source in the autopilot has an automatic transition in both directions. What’s the significance of this? Well, it makes the user interaction more vulnerable to error, because the user may get confused going in either direction—from manual to automatic or from automatic to manual operation.

And indeed, there have been several reported incidents in which pilots engaged the **VERTICAL NAVIGATION** mode and entered a series of new speed limits into the flight management computer, anticipating that the computer would change the speed automatically at specific waypoints during a descent. Then air traffic control instructs the crew, for example, to hold at a certain altitude. The pilot switches to **ALTITUDE HOLD**, but in his or her mind, still assumes that all the speed limits so laboriously entered into the computer will be honored. However, since the autopilot is not in **VERTICAL NAVIGATION** mode, the speed input no longer comes from the flight management computer. Instead, as we already know, engagement of the **ALTITUDE HOLD MODE** causes a transition to the flight control panel as the source for the speed reference value. But since there is no speed value there (the pilot did not enter anything), the autopilot defaults to the aircraft’s speed at the moment of mode switching, which, in most cases, is very different from the speed limits entered by the pilots into the computer. If the crew does not catch this discrepancy in time, it can easily result in a speed violation.

Chapter 15

This chapter is based on a preliminary report that was published on July 28, 1994, four weeks after the aircraft accident, as well as on several trade publications and articles on the topic. The accident report was reprinted in its entirety in the *Aviation Week and Space Technology* magazine (1995, volume 72, issues 14, 15, 16, 20, 21, 22). As mentioned in the chapter, the preliminary report leaves gaps in our understanding of the accident (the final report was never made publicly available). I have simplified some of the technical details of the autopilot, aerodynamics, and envelope protection system’s logic so as to make the chapter more readable. In describing the accident, I took the liberty of adding non-factual information from my own familiarity with automated aircraft.

Following the test-flight accident, many procedures and practices for flight tests were re-examined and changed. In addition, new certification rules for conducting test flights specify

that in order to test the **ALTITUDE CAPTURE** maneuver, the maneuver should only be attempted at an altitude of at least 8,000 feet, which provides ample altitude for recovery.

The problem of over-trust on automation became a research agenda in the late 1980s and early 1990s. Considerable work has been done to understand this problem in the context of aviation and medical systems. The problem, however, is not unique to medicine and aviation—it exists in every domain where automated control systems are supervised by humans. The interested reader can find additional information in the work of Kathleen Mosier and Linda Sitkaka as well as in a recent article by John Lee and Katrina See, “Trust in Automation: Designing for Appropriate Reliance” (which will be published in the *Human Factors Journal* in 2004). The importance of the decision to disengage or keep using an automated system is discussed in an article by Raja Parasurman and Victor Riley, “Humans and Automation: Use, Misuse, Disuse, Abuse,” which was mentioned in the endnotes for chapter 1.

Crew Resource Management is now a mature concept. When first introduced, it signaled a major shift in emphasis away from the maverick, individualistic pilot who has the “right stuff,” to a sensitive and well-attuned manager of a crew. The shift in emphasis was traumatic to many pilots who were brought up in the old school and could not adjust to the new. However, after 20 years, it is clear that a cultural change has taken place in the aviation industry, and indeed, it was for the best. The concept was initially dubbed as *cockpit* resource management, and was later modified to *crew*, when others, from surgical teams to maritime bridge officers and power-plant technicians, began adopting and using the concept in their training and operations. The initial work on Crew Resource Management was conducted by Hugh Ruffell Smith and John Lauber (see *Cockpit Resource Management* by Earl Wiener, Barbara Kanki, and Robert Helmreich, published by Academic Press, 1997).

A very emotional debate about the role of the computer and the pilot in controlling modern aircraft has been going on for years in the aviation community. The debate only intensifies around the topics of envelope protections and who (machine or pilot) has final authority. This debate has many implications beyond aviation, because automated machines are now implemented in many safety-critical systems, such as medical technology and nuclear power. There are several excellent publications and books on the human factors of cockpit automation: Earl Wiener and Renwick Curry, “Flight-deck Automation: Promises and Problems” (published in the journal *Ergonomics*, volume 23, in 1980), and Charles Billings’ *Aviation Automation: The Search for a Human-Centered Approach* (published in 1996 by Erlbaum). The most current book on the topic is Thomas Sheridan’s *Humans and Automation* (published in 2002 by John Wiley and Sons).

Chapter 16

Semiautomatic transmission systems, such as the one presented here, can be found in older buses and construction equipment. Modern transmission systems are somewhat more sophisticated: they have an automatic feature that down-shifts between modes. Think of this as a kind of protection system: if the user fails to down shift when the bus is already at the highest state (e.g., high-1) and the speed is still dropping, an automatic transition into **MEDIUM** will take place so as to prevent the vehicle from stalling. A similar transition is also present between **MEDIUM** and **LOW**. Note that it is practically impossible for the driver to predict the onset of such automatic transitions, because they are based on a variety of conditions that are hidden from the driver. In a situation like this, it is possible to provide the driver with some indication just before the automatic transition takes place. The driver can